

M7-L1 Problem 2

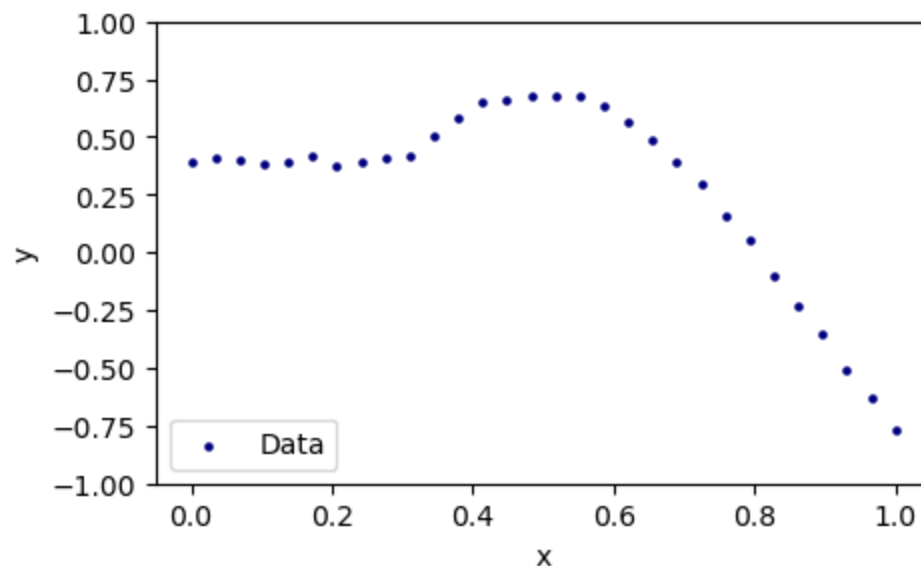
In this problem, you will explore what happens when you change the weights/biases of a neural network.

Neural networks act as functions that attempt to map from input data to output data. In training a neural network, the goal is to find the values of weights and biases that minimize the loss between their output and the desired output. This is typically done with a technique called backpropagation; however, here you will simply note the effect of changing specific weights in the network which has been pre-trained.

First, load the data and initial weights/biases below:

```
In [2]: import numpy as np
import matplotlib.pyplot as plt

x = np.array([0.          , 0.03448276, 0.06896552, 0.10344828, 0.13793103, 0.17241379, 0.20689655, 0.24137931, 0.27586207,
0.30981479, 0.34376751, 0.37772023, 0.41167295, 0.44562567, 0.47957839, 0.51353111, 0.54748383, 0.58143655, 0.61538927, 0.64934199,
0.68329471, 0.71724743, 0.75120015, 0.78515287, 0.81910559, 0.85305831, 0.88701103, 0.92096375, 0.95491647, 0.98886919, 1.02282191, 1.05677463, 1.09072735, 1.12468007, 1.15863279, 1.19258551, 1.22653823, 1.26049095, 1.29444367, 1.32839639, 1.36234911, 1.39630183, 1.43025455, 1.46420727, 1.49815999, 1.53211271, 1.56606543, 1.60001815, 1.63397087, 1.66792359, 1.70187631, 1.73582903, 1.76978175, 1.80373447, 1.83768719, 1.87163991, 1.90559263, 1.93954535, 1.97349807, 2.00745079, 2.04140351, 2.07535623, 2.10930895, 2.14326167, 2.17721439, 2.21116711, 2.24511983, 2.27907255, 2.31302527, 2.34697799, 2.38093071, 2.41488343, 2.44883615, 2.48278887, 2.51674159, 2.55069431, 2.58464703, 2.61859975, 2.65255247, 2.68650519, 2.72045791, 2.75441063, 2.78836335, 2.82231607, 2.85626879, 2.89022151, 2.92417423, 2.95812695, 2.99207967, 3.02603239, 3.05998511, 3.09393783, 3.12789055, 3.16184327, 3.19579599, 3.22974871, 3.26370143, 3.29765415, 3.33160687, 3.36555959, 3.39951231, 3.43346503, 3.46741775, 3.50137047, 3.53532319, 3.56927591, 3.60322863, 3.63718135, 3.67113407, 3.70508679, 3.73903951, 3.77299223, 3.80694495, 3.84089767, 3.87485039, 3.90880311, 3.94275583, 3.97670855, 4.01066127, 4.04461399, 4.07856671, 4.11251943, 4.14647215, 4.18042487, 4.21437759, 4.24833031, 4.28228303, 4.31623575, 4.35018847, 4.38414119, 4.41809391, 4.45204663, 4.48599935, 4.51995207, 4.55390479, 4.58785751, 4.62181023, 4.65576295, 4.68971567, 4.72366839, 4.75762111, 4.79157383, 4.82552655, 4.85947927, 4.89343199, 4.92738471, 4.96133743, 4.99529015, 5.02924287, 5.06319559, 5.09714831, 5.13110103, 5.16505375, 5.19900647, 5.23295919, 5.26691191, 5.30086463, 5.33481735, 5.36877007, 5.40272279, 5.43667551, 5.47062823, 5.50458095, 5.53853367, 5.57248639, 5.60643911, 5.64039183, 5.67434455, 5.70829727, 5.742250, 5.77620272, 5.81015544, 5.84410816, 5.87806088, 5.9120136, 5.94596632, 5.97991904, 6.01387176, 6.04782448, 6.0817772, 6.11572992, 6.14968264, 6.18363536, 6.21758808, 6.2515408, 6.28549352, 6.31944624, 6.35339896, 6.38735168, 6.4213044, 6.45525712, 6.48920984, 6.52316256, 6.55711528, 6.591068, 6.62502072, 6.65897344, 6.69292616, 6.72687888, 6.7608316, 6.79478432, 6.82873704, 6.86268976, 6.89664248, 6.9305952, 6.96454792, 6.99850064, 7.03245336, 7.06640608, 7.1003588, 7.13431152, 7.16826424, 7.20221696, 7.23616968, 7.2701224, 7.30407512, 7.33802784, 7.37198056, 7.40593328, 7.439886, 7.47383872, 7.50779144, 7.54174416, 7.57569688, 7.6096496, 7.64360232, 7.67755504, 7.71150776, 7.74546048, 7.7794132, 7.81336592, 7.84731864, 7.88127136, 7.91522408, 7.9491768, 7.98312952, 8.01708224, 8.05103496, 8.08498768, 8.1189404, 8.15289312, 8.18684584, 8.22079856, 8.25475128, 8.288704, 8.32265672, 8.35660944, 8.39056216, 8.42451488, 8.4584676, 8.49242032, 8.52637304, 8.56032576, 8.59427848, 8.6282312, 8.66218392, 8.69613664, 8.73008936, 8.76404208, 8.7979948, 8.83194752, 8.86589924, 8.89985196, 8.93380468, 8.9677574, 9.00171012, 9.03566284, 9.06961556, 9.10356828, 9.137521, 9.17147372, 9.20542644, 9.23937916, 9.27333188, 9.3072846, 9.34123732, 9.37518964, 9.40914236, 9.44309508, 9.4770478, 9.51099952, 9.54495224, 9.57890496, 9.61285768, 9.6468104, 9.68076312, 9.71471584, 9.74866856, 9.78262128, 9.816574, 9.85052672, 9.88447944, 9.91843216, 9.95238488, 9.9863376, 1.02028532, 1.05423804, 1.08819076, 1.12214348, 1.1560962, 1.19004892, 1.22400164, 1.25795436, 1.29190708, 1.3258598, 1.35981252, 1.39376524, 1.42771796, 1.46167068, 1.4956234, 1.52957612, 1.56352884, 1.59748156, 1.63143428, 1.665387, 1.69933972, 1.73329244, 1.76724516, 1.80119788, 1.8351506, 1.86910332, 1.90305604, 1.93700876, 1.97096148, 1.9999142, 2.03386692, 2.06781964, 2.10177236, 2.13572508, 2.1696778, 2.20363052, 2.23758324, 2.27153596, 2.30548868, 2.3394414, 2.37339412, 2.40734684, 2.44129956, 2.47525228, 2.509205, 2.54315772, 2.57711044, 2.61106316, 2.64501588, 2.6789686, 2.71292132, 2.74687404, 2.78082676, 2.81477948, 2.8487322, 2.88268492, 2.91663764, 2.95059036, 2.98454308, 3.0184958, 3.05244852, 3.08640124, 3.12035396, 3.15430668, 3.1882594, 3.22221212, 3.25616484, 3.29011756, 3.32407028, 3.358023, 3.39197572, 3.42592844, 3.45988116, 3.49383388, 3.5277866, 3.56173932, 3.59569204, 3.62964476, 3.66359748, 3.6975502, 3.73150292, 3.76545564, 3.79940836, 3.83336108, 3.8673138, 3.90126652, 3.93521924, 3.96917196, 4.00312468, 4.0370774, 4.07103012, 4.10498284, 4.13893556, 4.17288828, 4.206841, 4.24079372, 4.27474644, 4.30869916, 4.34265188, 4.3766046, 4.41055732, 4.44451004, 4.47846276, 4.51241548, 4.5463682, 4.58032092, 4.61427364, 4.64822636, 4.68217908, 4.7161318, 4.75008452, 4.78403724, 4.81798996, 4.85194268, 4.8858954, 4.91984812, 4.95380084, 4.98775356, 5.02170628, 5.055659, 5.08961172, 5.12356444, 5.15751716, 5.19146988, 5.2254226, 5.25937532, 5.29332804, 5.32728076, 5.36123348, 5.3951862, 5.42913892, 5.46309164, 5.49704436, 5.53099708, 5.5649498, 5.59890252, 5.63285524, 5.66680796, 5.70076068, 5.7347134, 5.76866612, 5.80261884, 5.83657156, 5.87052428, 5.904477, 5.93842972, 5.97238244, 6.00633516, 6.04028788, 6.0742406, 6.10819332, 6.14214604, 6.17609876, 6.21005148, 6.2440042, 6.27795692, 6.31190964, 6.34586236, 6.37981508, 6.4137678, 6.44772052, 6.48167324, 6.51562596, 6.54957868, 6.5835314, 6.61748412, 6.65143684, 6.68538956, 6.71934228, 6.753295, 6.78724772, 6.82119944, 6.85515216, 6.88910488, 6.9230576, 6.95701032, 6.99096304, 7.02491576, 7.05886848, 7.0928212, 7.12677392, 7.16072664, 7.19467936, 7.22863208, 7.2625848, 7.29653752, 7.33049024, 7.36444296, 7.39839568, 7.4323484, 7.46630112, 7.50025384, 7.53420656, 7.56815928, 7.602112, 7.63606472, 7.67001744, 7.70397016, 7.73792288, 7.7718756, 7.80582832, 7.83978104, 7.87373376, 7.90768648, 7.9416392, 7.97559192, 8.00954464, 8.04349736, 8.07745008, 8.1114028, 8.14535552, 8.17930824, 8.21326096, 8.24721368, 8.2811664, 8.31511912, 8.34907184, 8.38302456, 8.41697728, 8.45093, 8.48488272, 8.51883544, 8.55278816, 8.58674088, 8.6206936, 8.65464632, 8.68859904, 8.72255176, 8.75650448, 8.7904572, 8.82440992, 8.85836264, 8.89231536, 8.92626808, 8.9602208, 8.99417352, 9.02812624, 9.06207896, 9.09603168, 9.1299844, 9.16393712, 9.19788984, 9.23184256, 9.26579528, 9.299748, 9.33370072, 9.36765344, 9.40160616, 9.43555888, 9.4695116, 9.50346432, 9.53741704, 9.57136976, 9.60532248, 9.6392752, 9.67322792, 9.70718064, 9.74113336, 9.77508608, 9.8090388, 9.84299152, 9.87694424, 9.91089696, 9.94484968, 9.9788024, 1.00275512, 1.02670784, 1.05066056, 1.07461328, 1.098566, 1.12251872, 1.14647144, 1.17042416, 1.19437688, 1.2183296, 1.24228232, 1.26623504, 1.29018776, 1.31414048, 1.3380932, 1.36204592, 1.38599864, 1.40995136, 1.43390408, 1.4578568, 1.48180952, 1.50576224, 1.52971496, 1.55366768, 1.5776204, 1.60157312, 1.62552584, 1.64947856, 1.67343128, 1.697384, 1.72133672, 1.74528944, 1.76924216, 1.79319488, 1.8171476, 1.84110032, 1.86505304, 1.88900576, 1.91295848, 1.9369112, 1.96086392, 1.98481664, 2.00876936, 2.03272208, 2.0566748, 2.08062752, 2.10458024, 2.12853296, 2.15248568, 2.1764384, 2.20039112, 2.22434384, 2.24829656, 2.27224928, 2.296202, 2.32015472, 2.34410744, 2.36806016, 2.39201288, 2.4159656, 2.43991832, 2.46387104, 2.48782376, 2.51177648, 2.5357292, 2.55968192, 2.58363464, 2.60758736, 2.63154008, 2.6554928, 2.67944552, 2.70339824, 2.72735096, 2.75130368, 2.7752564, 2.79920912, 2.82316184, 2.84711456, 2.87106728, 2.89502, 2.91897272, 2.94292544, 2.96687816, 2.99083088, 3.0147836, 3.03873632, 3.06268904, 3.08664176, 3.11059448, 3.1345472, 3.15849992, 3.18245264, 3.20640536, 3.23035808, 3.2543108, 3.27826352, 3.30221624, 3.32616896, 3.35012168, 3.3740744, 3.39802712, 3.42197984, 3.44593256, 3.46988528, 3.493838, 3.51779072, 3.54174344, 3.56569616, 3.58964888, 3.6136016, 3.63755432, 3.66150704, 3.68545976, 3.70941248, 3.7333652, 3.75731792, 3.78127064, 3.80522336, 3.82917608, 3.8531288, 3.87708152, 3.90103424, 3.92498696, 3.94893968, 3.9728924, 3.99684512, 4.02079784, 4.04475056, 4.06870328, 4.092656, 4.11660872, 4.14056144, 4.16451416, 4.18846688, 4.2124196, 4.23637232, 4.26032504, 4.28427776, 4.30823048, 4.3321832, 4.35613592, 4.38008864, 4.40404136, 4.42799408, 4.4519468, 4.47589952, 4.49985224, 4.52380496, 4.54775768, 4.5717104, 4.59566312, 4.61961584, 4.64356856, 4.66752128, 4.691474, 4.71542672, 4.73937944, 4.76333216, 4.78728488, 4.8112376, 4.83519032, 4.85914304, 4.88309576, 4.90704848, 4.9310012, 4.95495392, 4.97890664, 5.00285936, 5.02681208, 5.0507648, 5.07471752, 5.09867024, 5.12262296, 5.14657568, 5.1705284, 5.19448112, 5.21843384, 5.24238656, 5.26633928, 5.290292, 5.31424472, 5.33819744, 5.36215016, 5.38610288, 5.4100556, 5.43400832, 5.45796104, 5.48191376, 5.50586648, 5.5298192, 5.55377192, 5.57772464, 5.60167736, 5.62563008, 5.6495828, 5.67353552, 5.69748824, 5.72144096, 5.74539368, 5.7693464, 5.79329912, 5.81725184, 5.84120456, 5.86515728, 5.88911, 5.91306272, 5.93701544, 5.96096816, 5.98492088, 6.0088736, 6.03282632, 6.05677904, 6.08073176, 6.10468448, 6.1286372, 6.15258992, 6.17654264, 6.20049536, 6.22444808, 6.2484008, 6.27235352, 6.29630624, 6.32025896, 6.34421168, 6.3681644, 6.39211712, 6.41606984, 6.44002256, 6.46397528, 6.487928, 6.51188072, 6.53583344, 6.55978616, 6.58373888, 6.6076916, 6.63164432, 6.65559704, 6.67954976, 6.70350248, 6.7274552, 6.75140792, 6.77536064, 6.79931336, 6.82326608, 6.8472188, 6.87117152, 6.89512424, 6.91907696, 6.94302968, 6.9669824, 6.99093512, 7.01488784, 7.03884056, 7.06279328, 7.086746, 7.11069872, 7.13465144, 7.15860416, 7.18255688, 7.2065096, 7.23046232, 7.25441504, 7.27836776, 7.30232048, 7.3262732, 7.35022592, 7.37417864, 7.39813136, 7.42208408, 7.4460368, 7.46998952, 7.49394224, 7.51789496, 7.54184768, 7.5657994, 7.58975212, 7.61370484, 7.63765756, 7.66161028, 7.685563, 7.70951572, 7.73346844, 7.75742116, 7.78137388, 7.8053266, 7.82927932, 7.85323204, 7.87718476, 7.90113748, 7.9250902, 7.94904292, 7.97299564, 7.99694836, 8.02090108, 8.0448538, 8.06880652, 8.09275924, 8.11671196, 8.14066468, 8.1646174, 8.18857012, 8.21252284, 8.23647556, 8.26042828, 8.284381, 8.30833372, 8.33228644, 8.35623916, 8.38019188, 8.4041446, 8.42809732, 8.45205004, 8.47599976, 8.49995248, 8.5239052, 8.54785792, 8.57181064, 8.59576336, 8.61971608, 8.6436688, 8.66762152, 8.69157424, 8.71552696, 8.73947968, 8.7634324, 8.78738512, 8.81133784, 8.83529056, 8.85924328, 8.883196, 8.90714872, 8.93110144, 8.95505416, 8.97900688, 9.0029596, 9.02691232, 9.05086504, 9.07481776, 9.09877048, 9.1227232, 9.14667592, 9.17062864, 9.19458136, 9.21853408, 9.2424868, 9.26643952, 9.29039224, 9.31434496, 9.33829768, 9.3622504, 9.38620312, 9.41015584, 9.43410856, 9.45806128, 9.482014, 9.50596672, 9.52991944, 9.55387216, 9.57782488, 9.6017776, 9.62573032, 9.64968304, 9.67363576, 9.69758848, 9.7215412, 9.74549392, 9.76944664, 9.79339936, 9.81735208, 9.8413048, 9.86525752, 9.88921024, 9.91316296, 9.93711568, 9.9610684, 9.98502112, 1.00897384, 1.03292656, 1.05687928, 1.080832, 1.10478472, 1.12873744, 1.15269016, 1.17664288, 1.2005956, 1.22454832, 1.24850104, 1.27245376, 1.29640648, 1.3203592, 1.34431192, 1.36826464, 1.39221736, 1.41617, 1.44012272, 1.46407544, 1.48802816, 1.51198088, 1.5359336, 1.55988632, 1.58383904, 1.60779176, 1.63174448, 1.6556972, 1.67964992, 1.70360264, 1.72755536, 1.75150808, 1.7754608, 1.79941352, 1.82336624, 1.84731896, 1.87127168, 1.8952244, 1.91917712, 1.94312984, 1.96708256, 1.99103528, 2.014988, 2.03894072, 2.06289344, 2.08684616, 2.11079888, 2.1347516, 2.15870432, 2.18265704, 2.20660976, 2.23056248, 2.2545152, 2.27846792, 2.30242064, 2.32637336, 2.35032608, 2.3742788, 2.39823152, 2.42218424, 2.44613696, 2.47008968, 2.4940424, 2.51799512, 2.54194784, 2.56590056, 2.58985328, 2.613806, 2.63775872, 2.66171144, 2.68566416, 2.70961688, 2.7335696, 2.75752232, 2.78147504, 2.80542776, 2.82938048, 2.8533332, 2.87728592, 2.90123864, 2.92519136, 2
```



MLP Function

Copy in your MLP function (and all necessary helper functions) below. Make sure it is called `MLP()`. In this case, you can plug in `x`, `weights`, and `biases` to try and predict `y`. Make sure you use the sigmoid activation function after each layer (except the final layer).

```
In [3]: # YOUR CODE GOES HERE
def perceptron_layer(x, weight, bias):
    # YOUR CODE GOES HERE
    return (np.dot(x,weight.T)) + bias

def sigmoid(x):
    return 1./(1.+np.exp(-x))

def MLP(x, weights, biases):
    # YOUR CODE GOES HERE
    y = x
    for i in range(len(weights)):
        y = perceptron_layer(y,weights[i],biases[i])
        if(i < len(weights) - 1):
            y = sigmoid(y)
    return y
```

Varying weights

The provided network has 2 hidden layers, each with 3 neurons. The weights and biases are shown below. Note the weights w_a and w_b -- these are left for you to investigate:

$$\underline{x}; (N \times 1) \rightarrow \sigma(w = \begin{bmatrix} -5.9 \\ \mathbf{w_a} \\ \mathbf{w_b} \end{bmatrix}; b = \begin{bmatrix} 2.02 \\ -3.48 \\ -1.12 \end{bmatrix}) \rightarrow \underline{(N \times 3)} \rightarrow \sigma(w = \begin{bmatrix} 0.9 & -1. & -1.65 \\ 4.76 & -0.89 & -2.93 \\ -0.95 & 3.19 & 2.61 \end{bmatrix}; b = \begin{bmatrix} 1.35 \\ -0.11 \\ -4.03 \end{bmatrix}) \rightarrow \underline{(N \times 3)} \rightarrow \sigma(w = \begin{bmatrix} 1.72 \\ -1.56 \\ -3.31 \end{bmatrix}; b = \begin{bmatrix} 0.52 \end{bmatrix}) \rightarrow \hat{y}; (N \times 1)$$

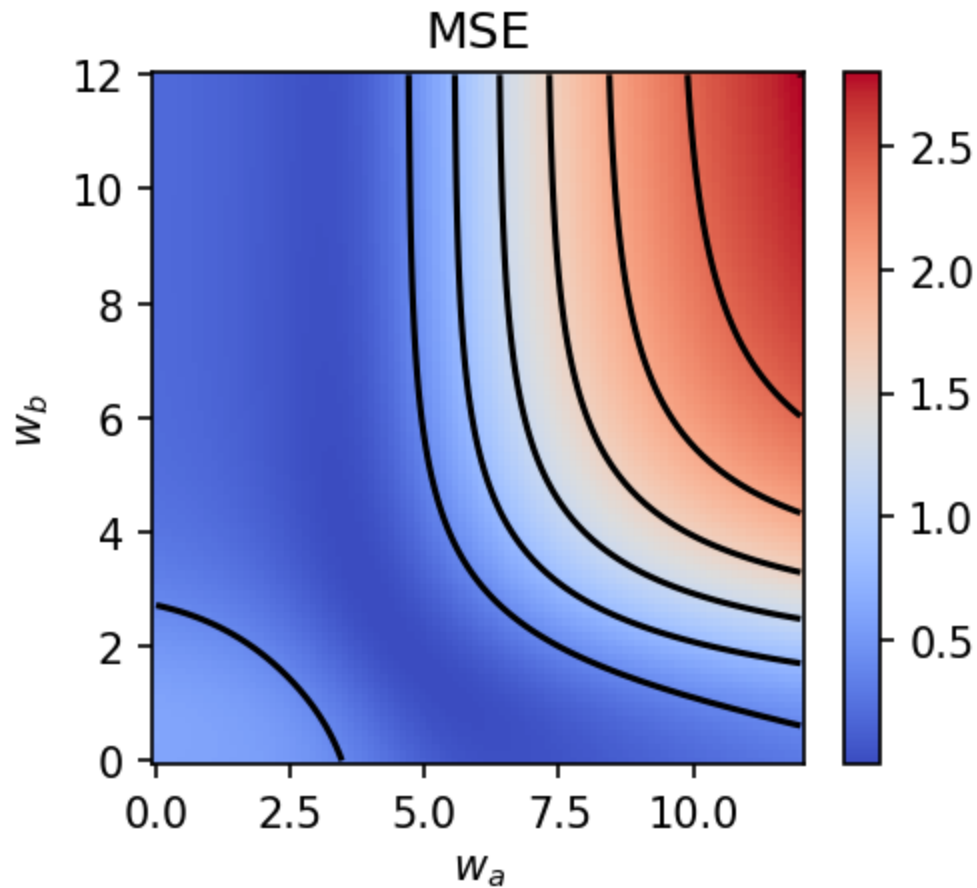
We can compute the MSE for each combination of (w_a, w_b) to see where MSE is minimized.

```
In [4]: def MSE(y, pred):
        return np.mean((y.flatten()-pred.flatten())**2)

vals = np.linspace(0,12,100)
was, wbs = np.meshgrid(vals,vals)
mses = np.zeros_like(was.flatten())

for i in range(len(was.flatten())):
    ws, bs = weights.copy(), biases.copy()
    ws[0][1,0] = was.flatten()[i]
    ws[0][2,0] = wbs.flatten()[i]
    mses[i] = MSE(y, MLP(x, ws, bs))
mses = mses.reshape(was.shape)

plt.figure(figsize = (3.5,3),dpi=150)
plt.title("MSE")
plt.contour(was,wbs,mses,colors="black")
plt.pcolormesh(was,wbs,mses,shading="nearest",cmap="coolwarm")
plt.xlabel("$w_a$")
plt.ylabel("$w_b$")
plt.colorbar()
plt.show()
```



```
In [5]: %matplotlib inline
from ipywidgets import interact, interactive, fixed, interact_manual, Layout, FloatSlider, Dropdown

def plot(wa, wb):
    ws, bs = weights.copy(), biases.copy()
    ws[0][1,0] = wa
    ws[0][2,0] = wb

    xs = np.linspace(0,1)
    ys = MLP(xs.reshape(-1,1), ws, bs)

    plt.figure(figsize=(10,4),dpi=120)
```

```

plt.subplot(1,2,1)
plt.contour(was,wbs,mse,color="black")
plt.pcolormesh(was,wbs,mse,shading="nearest",cmap="coolwarm")
plt.title(f"$w_a = {wa:.1f}$; $w_b = {wb:.1f}$")
plt.xlabel("$w_a$")
plt.ylabel("$w_b$")
plt.scatter(wa,wb,marker="*",color="black")
plt.colorbar()

plt.subplot(1,2,2)
plt.scatter(x,y,s=5,c="navy",label="Data")
plt.plot(xs,ys,"r-",linewidth=1,label="MLP")
plt.title(f"MSE = {MSE(y, MLP(x, ws, bs)):.3f}")
plt.legend(loc="lower left")
plt.ylim(-1,1)
plt.xlabel("x")
plt.ylabel("y")

plt.show()

slider1 = FloatSlider(
    value=0,
    min=0,
    max=12,
    step=.5,
    description='wa',
    disabled=False,
    continuous_update=True,
    orientation='horizontal',
    readout=False,
    layout = Layout(width='550px')
)

slider2 = FloatSlider(
    value=0,
    min=0,
    max=12,
    step=.5,
    description='wb',
    disabled=False,
    continuous_update=True,
    orientation='horizontal',
    readout=False,
    layout = Layout(width='550px')
)

```

```
)  
  
interactive_plot = interactive(  
    plot,  
    wa = slider1,  
    wb = slider2  
)  
output = interactive_plot.children[-1]  
output.layout.height = '500px'  
  
interactive_plot
```

Out[5]: interactive(children=(FloatSlider(value=0.0, description='wa', layout=Layout(width='550px'), max=12.0, readout...

Questions

1. For $w_a = 4.0$, what value of w_b gives the lowest MSE (to the nearest 0.5)?

- *ANSWER:* $w_b = 3.0$ gives the MSE of 0.001

1. For the large values of w_a and w_b , describe the MLP's predictions.

- *ANSWER:* The MLP's prediction gives a poor fitting curve. The MSE keeps increasing as it deviates more and more from the data.